REMARKS

I.   Introduction

Claims 28 to 42 are currently pending. In view of the following remarks, it is respectfully submitted that all of the presently pending claims are allowable, and reconsideration is respectfully requested.

II.   Allowable Subject Matter

Applicants thank the Examiner for indicating that claim 30 includes allowable subject matter. In this regard, the Examiner will note that claim 30 has been rewritten herein to be in independent form and to include all of the features of its base claim. It is therefore respectfully submitted that claim 30 in condition for immediate allowance.

III.   Rejection of Claim 28 Under 35 U.S.C. § 103(a)

Claim 28 was rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of U.S. Patent No. 6,088,788 ("Borkenhagen et al."), U.S. Patent No. 6,298,431 ("Gottlieb"), and U.S. Patent No. 6,138,230 ("Hervin et al."). It is respectfully submitted that the combination of Borkenhagen et al., Gottlieb, and Hervin et al. does not render unpatentable claim 28 for at least the following reasons.

To reject a claim under 35 U.S.C. § 103(a), the Office bears the initial burden of presenting a *prima facie* case of obviousness. *In re Rijckaert*, 9 F.3d 1531, 1532, 28 U.S.P.Q.2d 1955, 1956 (Fed. Cir. 1993). To establish *prima facie* obviousness, three criteria must be satisfied. First, there must be some suggestion or motivation to modify or combine reference teachings. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988). This teaching or suggestion to make the claimed combination must be found in the prior art and not based on the application disclosure. *In re Vaeck*, 947 F.2d 488, 20 U.S.P.Q.2d 1438 (Fed. Cir. 1991). Second, there must be a reasonable expectation of success. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 U.S.P.Q. 375 (Fed. Cir. 1986).   Third, the prior art reference(s) must teach or suggest all of the claim features. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974).

Claim 28 relates to an instruction pipeline in a microprocessor and recites, *inter alia*, the following:

> *. . . at least one of the plurality of pipeline units is configured to: receive an instruction from another of the pipeline unit; responsive to the receipt of the instruction: issue the received instruction to a downstream pipeline unit; and store a copy of the received instruction; and subsequent to the issuing of the received instruction,*

issue to the downstream pipeline unit the copy of the received instruction after a stall occurs in the one of the multiple threads.

The Office Action admits that neither Borkenhagen et al. nor Gottlieb disclose, responsive to the receipt of an instruction, issuing the received instruction to a downstream pipeline unit and storing a copy of the received instruction. The Office Action instead refers to Hervin et al., column 19, lines 28 to 49, as assertedly correcting this deficiency of the combination of Borkenhagen et al. and Gottlieb. However, the cited section of Hervin et al. refers to storing **a state** at a point prior to execution of steps that are based on a prediction of a conditional branch outcome. Storing a state includes storing data which may be altered by execution of instructions fetched based on the incorrectly predicted branch outcome. *See* Hervin et al., column 6, lines 47 to 60. While Hervin et al. may refer to storing a state, Hervin et al. do not disclose or suggest storing a copy of **an instruction** which is passed to a downstream pipeline unit. Instead, instructions are fetched and passed based on a branch prediction. If the prediction proves to be incorrect, then the previously fetched and passed instructions are flushed and <u>different</u> instructions are fetched and passed. *See, e.g.,* Hervin et al., column 6, lines 34 to 46. None of the cited references disclose or suggest using a checkpoint system, as provided in Hervin et al., for storing copies of received **instructions** that are passed to downstream pipeline units and/or for the purpose of flushing a pipeline, e.g., in the case of a thread switch. Indeed, even Hervin et al., which refers to a checkpoint system, discusses a pipeline flush and does <u>not</u> refer to use of the checkpoint system for performing the pipeline flush. *See, e.g.,* Hervin et al., column 5, lines 1 to 9, and column 6, lines 34 to 60. Accordingly, one skilled in the art would not further modify the system of Borkenhagen et al. to further include the features of issuing a received instruction and storing a copy of the received instruction based on Hervin et al.

In the "Response to Arguments" section, the Office Action conclusorily asserts that "Hervin discloses . . . copying an instruction by saving its context." However, saving a context is not equivalent to saving an instruction. Indeed, the point of the saving of the context is that an incorrect prediction may be made so that a ***different*** instruction would have to be used. Therefore, the context is saved for use with the different instruction.

In the "Response to Arguments" section, the Office Action further notes that Hervin indicates that the copying "'could be used for every instruction' (col 19 lines 42-44)." Final Office Action, page 13. However, the cited section of Hervin merely indicates that context saving can be done with respect to each instruction. The cited section does not disclose or suggest saving copies of the instructions themselves. None of the references,

whether considered alone or in combination, discloses or suggests responsive to receipt of an instruction, issuing the instruction downstream and also saving a copy of the instruction, as provided for in the context of claim 28.

Furthermore, prior art references must be considered as a whole, including portions that teach away from the claimed subject matter. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540 (Fed. Cir. 1983). Even with respect to storing the state at selective points in an execution sequence which may be incorrectly modified by execution of instructions fetched based on an incorrect conditional branch prediction, Hervin et al. note that use of checkpoint registers requires additional resources, and that a limit is placed on the extent to which the checkpoint functionality is provided in view of the trade-off between the required hardware and benefits of the checkpoint functionality. *See* Hervin et al., column 19, lines 42 to 49. Accordingly, Hervin et al. teach away from providing checkpoint memory structures for storing numerous pipeline instructions themselves (aside from the storing of a context) in order to flush a pipeline, as suggested by the Office Action.

In the "Response to Arguments" section, the Office Action dismisses the above argument based on a mischaracterization of that which is stated in In re Gurley, 27 F.3d 551, 553, 31 USPQ2d 1130 (Fed. Cir. 1994). In this regard, the Office Action asserts that In re Gurley states that the standard for teaching away is where "[a] reference . . . states that something cannot be done." Nowhere does In re Gurley make any such statement, and indeed, the law is not as stated by the Office Action. Instead, In re Gurley states that "[a] reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference." Upon reading Hervin, one skilled in the art would be discouraged from providing checkpoint memory structures for storing numerous pipeline instructions themselves (aside from the storing of a context) in order to flush a pipeline. Therefore, Hervin teaches away from this feature.

Furthermore, even assuming for argument's sake, that one would further modify the combination of Borkenhagen et al. and Gottlieb, based on Hervin et al., to include registers in which to store copies of instructions for the purpose of pipeline flushing (which one would not as explained above), the modified system would still not disclose or suggest the features of claim 28. Instead, at most, the modified system would provide that when a pipeline unit receives an instruction to execute, the pipeline unit which will execute the instruction stores a copy of the instruction in a checkpoint register so that, in case of a flush for a thread switch, the instruction can be restored when the original thread is continued. The

cited references do not disclose or suggest <u>a pipeline unit which both passes an instruction to</u> <u>a downstream pipeline unit</u> and stores a copy of the instruction, so that modification of the system of Borkenhagen et al. as suggested by the Office Action would not have been predictable without an improper hindsight reconstruction based on Applicants' disclosure.

Accordingly, the combination of Borkenhagen et al., Gottlieb, and Hervin et al. does not disclose or suggest all of the features recited in claim 28, so that the combination of Borkenhagen et al., Gottlieb, and Hervin et al. does not render unpatentable claim 28.

Withdrawal of this rejection is therefore respectfully requested.

IV.   <u>Rejection of Claims 29, 31 to 36, and 38 to 42 Under 35 U.S.C. § 103(a)</u>

Claims 29, 31 to 36, and 38 to 42[1] were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Borkenhagen et al., Gottlieb, Hervin et al., and U.S. Patent No. 5,907,702 ("Flynn et al."). It is respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable any of claims 29 to 36, and 38 to 42 for at least the following reasons.

Claim 29 recites, *inter alia*, the following:

> *. . . an instruction queue, wherein: in a first operating mode, the instruction queue is configured to, for each of the series of instructions, responsive to receipt of the instruction, pass the instruction from the at least one upstream pipeline unit to the at least one downstream pipeline unit . . . and store a copy of the instruction, . . . and in a second operating mode the instruction queue is configured to issue to the at least one downstream pipeline unit at least one of the copies on the one of the plurality of instruction threads on which a corresponding at least one of the series of instructions was previously issued.*

The Office Action asserts that Figure 2, queues 10 and 14 of Flynn et al. disclose an instruction queue configured to pass an instruction from an upstream pipeline unit to a downstream pipeline unit and to store the instruction. *See* Office Action, paragraph 5. However, claim 29 recites passing ***the instruction*** and storing a ***copy of the instruction***. As set forth in Applicants' previously filed Responses, the queues 10 and 14 store each received instruction and then pass on each of the *stored* instructions. Further, once the stored

---

[1]   While the rejection heading indicates that claim 30 is included in this rejection, the substantive arguments do not address claim 30. Indeed, the Office Action states that claim 30 would be allowable if rewritten in independent form.

instruction is passed, it is not indicated in Flynn et al. that the passed stored instruction is maintained so that it can be passed again to a downstream unit.

Furthermore, as set forth in Applicants' previously filed Responses, the queues 10 and 14 transmit instructions in a single manner, *i.e.*, they receive an instruction, store it in a queue, and transmit the queued instruction on to a downstream unit. They do not issue ***a received instruction*** and also ***a copy of the same received instruction***, as provided for in the context of claim 29.

Thus, even if the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. suggest a queue that receives an instruction, stores a copy of the instruction, transmits the copy to the downstream unit, receives another copy of the previously received instruction, and repeats the storing and transmitting steps with respect to the second copy (which Applicants do not concede), the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. still does not disclose or suggest the instruction queue of claim 29, which is configured to, responsive to receipt of an instruction, pass the received instruction and store a copy of the received instruction, and which is further configured to issue the stored copy. Instead, at most, it might be argued that the combination of the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. refers to storing a copy of a first received instruction and a copy of a second received instruction that corresponds to the first received instruction, e.g., where two different copies of the same instruction are received from an upstream unit, and transmitting to a downstream unit each of the stored copies.

Accordingly, the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not disclose or suggest all of the features recited in claim 29. It is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable claim 29.

Claim 32 recites, *inter alia*, the following:

> . . . *storing a copy of the issued original instruction in a queue; and passing the issued original instruction to a downstream unit on the one of the plurality of instruction threads; . . . and after detecting the stall, issuing at least one of the copies from the queue, on the one of the plurality of instruction threads on which the instructions were issued.*

As set forth above in support of the patentability of claim 29, the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not disclose or suggest these features. Since the combination of Borkenhagen et al., Gottlieb, Hervin et al., and

Flynn et al. does not disclose or suggest all of the features recited in claim 32, it is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable claim 32.

Claim 36 recites, *inter alia*, the following:

> *. . . an instruction queue configured to receive the instruction from the upstream pipeline unit, pass the received instruction to a downstream pipeline unit on the selected one of the plurality of threads, and store a copy of the received instruction, the instruction queue further configured to transmit, on the selected one of the plurality of threads, the copy of the received instruction in an event of a downstream stall on the selected one of the plurality of threads.*

As set forth above in support of the patentability of claim 29, the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not disclose or suggest these features. Since the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not disclose or suggest all of the features recited in claim 36, it is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable claim 36.

Claim 31 depends from claim 29 and therefore includes all of the features recited in claim 29. It is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable this dependent claim for the same reasons set forth above in support of the patentability of claim 29. *In re Fine, supra* (any dependent claim that depends from a non-obvious independent claim is non-obvious).

Claims 33 to 35 depend from claim 32 and therefore include all of the features recited in claim 32. It is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable these dependent claims for the same reasons set forth above in support of the patentability of claim 32. *Id*.

Claims 38 to 42 depend from claim 36 and therefore include all of the features recited in claim 36. It is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al. does not render unpatentable these dependent claims for the same reasons set forth above in support of the patentability of claim 36. *Id*.

In view of all of the foregoing, withdrawal of this rejection is respectfully requested.

V.   Rejection of Claim 37 Under 35 U.S.C. § 103(a)

Claim 37 was rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Borkenhagen et al., Gottlieb, Hervin et al., Flynn et al., and U.S. Patent No. 5,381,533 ("Peleg et al."). It is respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., Flynn et al., and Peleg et al. does not render unpatentable claim 37 for at least the following reasons.

Claim 37 depends from claim 36 and therefore includes all of the features recited in claim 36. Since Peleg et al. do not cure the deficiencies noted above with respect to the combination of Borkenhagen et al., Gottlieb, Hervin et al., and Flynn et al., it is therefore respectfully submitted that the combination of Borkenhagen et al., Gottlieb, Hervin et al., Flynn et al., and Peleg et al. does not render unpatentable this dependent claim for the same reasons set forth above in support of the patentability of claim 36. *Id.*
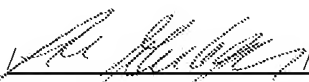
Withdrawal of this rejection is therefore respectfully requested.

VI.   Conclusion

In light of the foregoing, it is respectfully submitted that all of the presently pending claims are in condition for allowance. Prompt reconsideration and allowance of the present application are therefore earnestly solicited.

Respectfully submitted,

Dated: June 17, 2008   By: /Aaron Grunberger/
                                        Aaron Grunberger
                                        Reg. No. 59,210

                                        KENYON & KENYON LLP
                                        One Broadway
                                        New York, New York 10004
                                        (212) 425-7200

                                        CUSTOMER NO 26646